Phrase Detection in the Wikipedia

Miro Lehtonen[1] and Antoine Doucet[1,2]

[1] Department of Computer Science
P. O. Box 68 (Gustaf Hällströmin katu 2b)
FI–00014 University of Helsinki
Finland
{Miro.Lehtonen,Antoine.Doucet} @cs.helsinki.fi
[2] GREYC CNRS UMR 6072,
University of Caen Lower Normandy
F-14032 Caen Cedex
France
Antoine.Doucet @info.unicaen.fr

Abstract. The Wikipedia XML collection turned out to be rich of marked-up phrases as we carried out our INEX 2007 experiments. Assuming that a phrase occurs at the inline level of the markup, we were able to identify over 18 million phrase occurrences, most of which were either the anchor text of a hyperlink or a passage of text with added emphasis. As our IR system — EXTIRP — indexed the documents, the detected inline-level elements were duplicated in the markup with two direct consequences: 1) The frequency of the phrase terms increased, and 2) the word sequences changed. Because the markup was manipulated before computing word sequences for a phrase index, the actual multi-word phrases became easier to detect. The effect of duplicating the inline-level elements was tested by producing two run submissions in ways that were similar except for the duplication. According to the official INEX 2007 metric, the positive effect of duplicated phrases was clear.

1 Introduction

In previous years, our INEX-related experiments have included two dimensions to phrase detection, one at the markup level [1] and another in the term sequence analysis [2]. The methods have been tested on plain text corpora and scientific articles in XML format. The Wikipedia XML documents are the first collection of hypertext documents where our phrase detection methods are applied.

Regarding marked-up phrases, the nature of the markup in a hypertext document differs from that in a scientific article. The phrases that are marked in scientific texts are mostly meant to be displayed with a different typeface, e.g. italicised or underlined, whereas hypertext documents have similar XML structures for marking the anchor text related to a hyperlink. Both emphasised passages and anchors are important, but whether they can be treated equally is still an open question. The initial results support the idea that emphasised phrases and anchors are equal as long as they are marked with similar XML structures — inline-level elements.

This article is organised as follows. Section 2 describes our IR system as it was implemented in 2007. In Section 3, we go through the phrase detection process step by step, from the original XML fragment to an intermediate XML format and, further, to the vector representation. The observations of the inline elements in the actual test collections are summarised in Section 4. How we extracted multiword units this year is explained in Section 5. Our results are presented in Section 6, and finally, we draw conclusions and directions for future work in Section 7.

2 EXTIRP baseline

The EXTIRP baseline without duplicated phrases is similar to our INEX 2006 submission [3] except for a few major bugs that have been fixed. The results are thus not comparable. First, EXTIRP scans through the document collection and selects disjoint fragments of XML to be indexed as atomic units. Typical fragments include XML elements marking sections, subsections, and paragraphs. In the Wikipedia, typical names for these elements are `article`, `section`, and `p`. The disjoint fragments are treated as traditional documents which are independent of each other. The pros include that the traditional IR methods apply, so we use the vector space model with a weighting scheme based on the tf*idf. The biggest of the cons is that the size of the indexed fragments is static, and if bigger or smaller answers are more appropriate for some query, the fragments have to be either divided further or combined into bigger fragments.

Second, two separate inverted indices are built for the fragments. A *word index* is created after punctuation and stopwords are removed and the remaining words are stemmed with the Porter algorithm [4]. The *phrase index* is based on Maximal Frequent Sequences (MFS) [5]. A sequence is said to be frequent if it occurs more often than a given sentence frequency threshold. It is said to be maximal if no other word can be inserted into the sequence without reducing the frequency below the threshold. This permits to obtain a compact set of document descriptors, that we use to build a phrase index of the collection. The frequency threshold is decided experimentally, because of the computational complexity of the algorithm. Although lower values for the threshold produce more MFSs, the computation itself would take too long to be practical. For the wikipedia collection, we used a frequency threshold of seven.

When processing the queries, we compute the cosine similarity between the document and the base term vectors which results in a `Word_RSV` value. In a similar fashion, each fragment vector gets a similarity score `MFS_RSV` for phrase similarity. These two scores are aggregated into a single RSV so that the aggregated RSV = α * `Word_RSV` + β * `MFS_RSV`, where α is the number of distinct query terms and β is the number of distinct query terms in the query phrases.

3 Phrase detection and duplication

The novelty in our system of 2007 was the analysis of the XML structure in order to locate marked-up phrases in the content. Table 1 shows an example of a passage in an XML fragment with two phrases marked up: “Britney Spears” and “I’m A Slave 4 U”. Both of the phrases are presumably better descriptors of the passage than any of the other nouns, e.g. “success” or “single”, which is also indicated by the corresponding document frequencies.

```
He repeated the success by doing the same with
    <c>Britney Spears</c>' dance-pop single,
       &quot;<c>I'm A Slave 4 U</c>&quot;
```

Table 1. A passage before phrase detection. The tag name `c` is an abbreviation of `collectionlink`. Link references are omitted.

3.1 Definition of qualified inline elements

Because there are many XML elements in the document that have a similar structure but a very different function, we need a formal definition for the kind of elements that most likely contain a marked-up phrase. Therefore, we define a *Qualified inline element* as follows: An XML element is considered a qualified inline element when the corresponding element node in the document tree meets the following conditions:

(1) The text node siblings contain at least n characters after whitespace has been normalised.
(2) The text node descendants contain at least m characters after normalisation.
(3) The element has no element node descendants.
(4) The element content is separated from the text node siblings by word delimiters, e.g. whitespace or commas.

When the whitespace of a text node is normalised, all the leading and trailing whitespace characters are trimmed off. We set the parameters to a minimum of three (3) characters in at least one text node child and a minimum of five (5) characters in at least one text node sibling, so that $n = 5$ and $m = 3$. With this definition, we disqualify those inline-level elements that 1) only contain one or two characters, and 2) those that contain several sentences of text, and 3) those that contain other XML elements. Defining the lower bounds of n and m improves the quality of detected phrases in the qualified inline elements. However, regarding the effectiveness of IR, the benefit of setting the parameters is marginal: very short character strings are usually ignored, whereas several sentences of text rarely match any searched phrase.

3.2 Doubling "Britney Spears"

According to our hypothesis, whatever is emphasised in the document should also be emphasised in the index. Consequently, all the occurrences of qualified inline elements are duplicated before the text is indexed. An example of such duplication is shown in Table 2 which represents the intermediate XML format which is the basis for the eventual vector representation.

```
        He repeated the success by doing the same with
<c>Britney Spears</c> <c>Britney Spears</c>' dance-pop single,
  &quot;<c>I'm A Slave 4 U</c> <c>I'm A Slave 4 U</c>&quot;
```

Table 2. The passage after phrase detection and duplication.

The example is representative of the most common appearances of "Britney Spears", which include the following:

```
freq.  appearance
-----  ----------
447    <collectionlink>Britney Spears</collectionlink>
 12    <emph2>Britney</emph2>
  5    <collectionlink>Britney</collectionlink>
```

After stemming and stopword removal, the corresponding word sequence becomes

```
britnei spear britnei spear
```

which is the input when extracting Maximal Frequent Sequences. Obviously, duplication has a dual effect of both increasing the term frequencies of the content that it concerns and changing the word sequence that phrases are extracted from. We believe the increase in term frequency is good because double "Britney Spears" is easier to spot than a single occurrence. The newly modified word sequence is also better as the MFS's that we extract also include the phrase `spear britnei` which, in addition to `britnei spear`, contributes to the score for phrase similarity (`MFS_RSV`). Duplicating phrases with more than two word units has a similar effect, as any word permutation within the phrase contributes to the `MFS_RSV` score.

4 Qualified inline elements in the Wikipedia XML

The most common elements that were duplicated are summarised in Table 3. The exhaustivity of an element type is the percentage of element occurrences duplicated out of all occurrences of that element.

XML Element	Count	Exhaustivity %	Percentage
collectionlink	12,971,384	76.2	69.1
unknownlink	2,372,870	60.0	12.6
emph2	1,339,345	49.2	7.1
emph3	992,373	67.0	5.3
p	282,438	10.3	1.5
outsidelink	230,675	26.8	1.2
title	222,917	14.0	1.2
languagelink	114,828	14.5	0.6
emph5	57,443	70.8	0.3
wikipedialink	42,009	23.8	0.2
All links	15,734,890	68.9	83.8
All emphasis	2,406,372	55.3*	12.8
Total	18,784,132	35.7	100

Table 3. Distribution of the most frequent qualified inline elements by element type. *All element types marking emphasis might not be included in the figures.

Most of the qualified inline elements are links (83.8%) and only a minority mark emphasis (12.8%) in the Wikipedia XML collection, which is the opposite of the collection of IEEE Computer Society[3] journals and transactions where the share of links is only 2.0% while 85.0% mark emphasis. The frequency of qualified inline elements is bigger in the Wikipedia in general, as well: 35.7% of all elements meet the requirements, whereas the corresponding figure is 6.6% in the IEEE collection.

5 MFS extraction

In this section, we are comparing our runs from the point of view of the MFSs that were extracted. We conjecture that the phrase duplication process facilitates the extraction of the more useful sequences, hereby inducing better retrieval performance. We will try to confirm this by analysing the extracted sequence sets corresponding to our runs.

Statistics are summarized in Table 4. As discussed earlier, the frequency threshold was always seven occurrences. That is, a sequence was considered frequent if it occurred in at least seven minimal units of a same document cluster. In the first run (UHel-Run1), we split the XML fragments extracted from the document collection into 500 disjoint clusters, whereas for UHel-Run2, the number of clusters is 250. Given the constant frequency threshold, a lower number of clusters causes a slower extraction but naturally permits finding more MFS occurrences. This is because it is easier to find seven occurrences of an MFS in larger clusters, that is, when the number of disjoint clusters is smaller [6].

[3] http://www.computer.org/

Run	Clusters	Number of sequences (total freq)	Average length	Average Frequence
UHel-Run1	500	21,009,668	2.248	19.9
UHel-Run2	250	37,252,061	2.184	26.4

Table 4. Per run statistics of the extracted MFS sets (frequency threshold: 7).

To give a first hint on the benefit of our phrase duplication technique, we are displaying the 10 most frequent phrases that were duplicated in Table 5.

Frequency	Phrase
37,474	Native American
37,328	population density
37,047	African American
36,046	married couples
35,926	per capita income
35,829	other races
35,807	poverty line
35,764	Pacific Islander
32,974	United States Census Bureau
26,572	United States

Table 5. The 10 most frequent phrases that were duplicated.

6 Results

We submitted two runs for the adhoc track task of Focused retrieval. The results are shown in Table 6. The assessments of 107 topics are included in the evaluation. The performance of our systems is relatively low compared with other evaluated systems, but the level seems typical of systems using tfidf-based weighting.

What we learn from these results is that our second run is undeniably better than the first run at all recall levels. The p values of the one-tailed t-test show that Run 2 is significantly better than Run 1 overall as well as at the lowest recall levels (0.00 and 0.01), given the threshold of 0.05. It is thus not only “Britney Spears” that is easier to find when doubled, but many other phrases that were topic titles. Although the EXTIRP baseline has a relatively low performance, it has been stable the past few years, and any improvement over its performance is hardly coincidence. We believe therefore that also other systems would benefit of the phrase extraction as we have done it.

	Run1		Run2			t-test	Best official
Recall level	Rank	Score	Rank	Score	Improvement	p	Score
MAiP	53	0.0912	45	0.1024	12.3%	0.0107	0.2238
0.00	66	0.3639	60	0.4068	11.8%	0.0454	0.6056
0.01	63	0.3319	58	0.3773	13.7%	0.0287	0.5271
0.05	58	0.2729	56	0.3000	9.9%	0.0783	0.4697
0.10	58	0.2273	54	0.2447	7.6%	0.1386	0.4234

Table 6. Performance of submissions “UHel-Run1” and “UHel-Run2” measured with interpolated precision at four recall levels. A total of 79 submissions are included in the ranking.

7 Conclusion

Phrase detection in the Wikipedia XML documents was a success as it improved our results at all recall levels. Analysing the XML markup did not involve any information about the document type, such as element names or tag names, so the technique is applicable to any XML documents. It can also be adopted by different systems as it is not tied to any specific document model or weighting method.

Our future work starts with the exploration of other algorithms for phrase extraction than the Maximal Frequent Sequences as we expect the duplication of inline elements to improve phrase extraction regardless of the algorithm. Another area of future development concerns the term weighting and matching in our system. We are interested in the effect of the phrase detection in more advanced and better performing systems, so we plan to discard the tfidf-based weights and move on to new directions.

References

1. Lehtonen, M.: Preparing heterogeneous XML for full-text search. ACM Trans. Inf. Syst. **24** (2006) 455–474
2. Doucet, A., Ahonen-Myka, H.: Probability and expected document frequency of discontinued word sequences, an efficient method for their exact computation. Traitement Automatique des Langues (TAL) **46** (2006) 13–37
3. Lehtonen, M., Doucet, A.: Extirp: Baseline retrieval from wikipedia. In Malik, S., Trotman, A., Lalmas, M., Fuhr, N., eds.: Comparative Evaluation of XML Information Retrieval Systems. Volume 4518 of Lecture Notes in Computer Science., Springer (2007) 119–124
4. Porter, M.F.: An algorithm for suffix stripping. Program **14** (1980) 130–137
5. Ahonen-Myka, H.: Finding all frequent maximal sequences in text. In Mladenic, D., Grobelnik, M., eds.: Proceedings of the 16th International Conference on Machine Learning ICML-99 Workshop on Machine Learning in Text Data Analysis, Ljubljana, Slovenia, J. Stefan Institute (1999) 11–17
6. Doucet, A., Ahonen-Myka, H.: Fast extraction of discontiguous sequences in text: a new approach based on maximal frequent sequences. In: Proceedings of IS-LTC 2006. (2006) 186–191